

Міністерство освіти і науки України  
Харківський національний університет імені В. Н. Каразіна  
Фізичний факультет  
Кафедра вищої математики

**«ЗАТВЕРДЖУЮ»**

Проректор з науково-педагогічної роботи

Пантелеймонов А.В.

«\_\_\_\_\_» \_\_\_\_\_ 2021 р.

## РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

### Інформатика та програмування/Математичне моделювання

(назва навчальної дисципліни)

рівень вищої освіти перший(бакалаврський)

галузь знань 10 Природничі науки

(шифр і назва)

спеціальність 104 Фізика та астрономія

(шифр і назва)

освітня програма Фізика, астрономія

(шифр і назва)

спеціалізація Фізика

(шифр і назва)

вид дисципліни обов'язкова

(обов'язкова /за вибором)

факультет фізичний

(назва факультету)

2021 / 2022 навчальний рік

Програму рекомендовано до затвердження Вченою радою фізичного факультету

“ 25 ” червня 2021 року, протокол № 8

РОЗРОБНИКИ ПРОГРАМИ: Завгородня Н.М., доцент, к.пед.н.

Програму схвалено на засіданні кафедри вищої математики фізичного факультету

Протокол від “ 30 ” серпня 2021 року № 1 \_\_\_\_\_

Завідувач кафедри вищої математики

\_\_\_\_\_ Завгородня Н.М.  
(підпис) (прізвище та ініціали)

Програму погоджено з гарантом освітньої (професійної/наукової) програми (керівником проектної групи) \_\_\_\_\_  
«фізика», «астрономія»  
назва освітньої програми

Гарант освітньої (професійної/наукової) програми (керівник проектної групи)

\_\_\_\_\_ Лазоренко О.В. \_\_\_\_\_  
(прізвище та ініціали) (підпис)

Програму погоджено методичною комісією фізичного факультету \_\_\_\_\_

Протокол від “ 31 ” серпня 2021 року № 1

Голова методичної комісії \_\_\_\_\_

\_\_\_\_\_ Макаровський М.О.  
(підпис) (прізвище та ініціали)

## ВСТУП

Програма навчальної дисципліни “ Інформатика та програмування / Математичне моделювання ” складена відповідно до освітньо-професійної (освітньо-наукової) програми підготовки

бакалаврів  
(назва рівня вищої освіти, освітньо-кваліфікаційного рівня)

спеціальності (напрямку) 6.040203 – фізика

спеціалізації

### 1. Опис навчальної дисципліни

1.1. Метою викладання навчальної дисципліни - навчити студентів основам програмування шляхом набуття вміння та навичок програмувати сучасною мовою програмування Python. Ознайомити з основами математичного моделювання та застосуванням відповідних методів на практиці при побудові алгоритмів та програм для розв’язку прикладних фізичних завдань.

1.2. Основні завдання вивчення дисципліни - отримання необхідних теоретичних знання та практичних навичок, що дозволять студентам розв’язувати з допомогою комп’ютера алгоритмічні задачі, будувати сучасні програмні продукти, вивчення мови програмування Python , тощо.

1.3. Кількість кредитів 5

1.4. Загальна кількість годин 150

1.5. Характеристика навчальної дисципліни	
Нормативна / за вибором	
Денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
2-й	
Семестр	
3-й	
Лекції	
32 год.	
Практичні, семінарські заняття	
Лабораторні заняття	
48 год.	
Самостійна робота	
69 год.	
Індивідуальні завдання 1	
	год.

### 1.6. Заплановані результати навчання.

У результаті вивчення навчальної дисципліни студент повинен

**знати:** основні поняття інформатики – виконавець, алгоритм, програма; синтаксис, семантику та властивості структур керування (ланцюг, розгалуження, цикли); синтаксичні конструкції мови програмування Python; формальні методи побудови алгоритмів та програм за допомогою рекурентних співвідношень; основні властивості програм; будову простих та складених типів даних; опис та використання підпрограм; поняття про виключні ситуації; роботу з файлами; модульне програмування; об'єктно-орієнтовне програмування.

**вміти:** будувати лінійні, розгалужені та циклічні алгоритми з використанням підпрограм, модулів та об'єктно-орієнтованого підходу у мові програмування Python.

## 2. Тематичний план навчальної дисципліни

### Розділ 1. Алгоритми і базові поняття програмування.

#### Тема 1. Алгоритми.

Змістовне поняття алгоритму. Основні властивості алгоритмів. Виконавець алгоритмів та його система команд. Абстракція даних. Команди управління. Базові управляючі структури. Формалізація поняття алгоритму.

#### Тема 2. Мови програмування.

Мови програмування (МП) як формальні мови описів алгоритмів. Структура МП. Синтаксис і семантика описів алгоритмів. Формальні засоби описів синтаксису МП. Функціональна і операційна семантика МП. МП Fortran та Python.

### Розділ 2. Основні структури програмування мовою Python.

#### Тема 3. Мова програмування Python.

Оператори управління МП. Складений оператор. Оператори вибору: типи, синтаксис і семантика. Техніка програмування розгалужень. Порівняння з МП Fortran.

#### Тема 4. Оператори управління МП Python.

Оператори управління МП. Оператори повторення: типи, синтаксис і семантика. Техніка програмування циклів. Передумови, постумови і інваріанти циклів. Поняття доведення програм. Методологія процедурного програмування.

#### Тема 5. Функції.

Структуризація алгоритму в термінах процедур і функцій. Синтаксис описів і семантика виконання. Організація обміну даними між процедурами. Локалізація даних. Техніка та програмування в термінах процедур і функцій. Відмінності в програмуванні процедур МП Fortran та Python.

#### Тема 6. Рекурсивні описи функцій.

Рекурсивні описи процедур і функцій. Індуктивне тлумачення рекурсивних описів функцій. Тлумачення через найменшу нерухому точку. Схеми для рекурсивного опису функцій.

#### Тема 7. Форми рекурсії.

Форми рекурсії. Лінійна рекурсія. Повторна рекурсія. Каскадна рекурсія. Удалена рекурсія. Взаємна рекурсія. Рекурсивні та ітеративні описи алгоритмів.

### Тема 8. Модулі.

Поняття модулів. Основи та особливості модульного програмування. Параметри модулів.

## Розділ 3. Структури. Елементи математичного моделювання.

### Тема 8. Структури даних.

Статичні типи даних: регулярний тип даних. Масиви. Одномірні масиви. Багатомірні масиви і загальні типи індексів. Динамічні і гнучкі масиви. Задачі обробки масивів. Множини. Скінченні множини як обчислювальні структури. Комбінований тип даних. Записи.

### Тема 10. Введення –виведення. Файли. Стандартна бібліотека.

Введення –виведення. Файли. Виключення. Обробка виключень. Стандартна бібліотека.

### Тема 11. Графи. Алгоритми обробки графів.

Графи. Алгоритми обробки графів. Бінарні дерева. Швидкі алгоритми сортування.

## 3. Структура навчальної дисципліни

Назви змістовних модулів і тем.	Кількість годин					
	Денна форма					
	усього	У тому числі				
л		п	лаб.	інд.	с. р.	
1	2	3	4	5	6	7
<b>Розділ 1. Алгоритми і базові поняття програмування.</b>						
Тема 1. Алгоритми.	6	2		2		2
Тема 2. Мови програмування.	12	2		4		6
<b>Разом за розділом 1</b>	18	4		6		8
<b>Розділ 2. Основні структури програмування мовою Python.</b>						
Тема 3. Мова програмування Python. Константи, змінні, типи даних. Оператори і вирази.	13	4		4		7
Тема 4. Оператори управління МП Python.	20	6		6		8
Тема 5. Функції	14	4		6		4
Тема 6. Рекурсивні описи функцій.	16	2		6		8
Тема 7. Форми рекурсії.	14	2		4		8
Тема 8. Модулі	19	4		8		7
<b>Разом за розділом 2</b>	96	22		34		42
<b>Розділ 3. Структури. Елементи математичного моделювання.</b>						
Тема 9. Структури даних. Множини. Класи. Методи. Спадкування.	14	4		4		6
Тема 10. Введення-	10	2		2		6

виведення. Файли. Стандартна бібліотека					
<b>Тема 11.</b> Графи. Бінарні дерева. Швидкі алгоритми сортування	12	2		2	8
<b>Разом за розділом 3.</b>	36	8		8	20
<b>Всього годин</b>	150	32		48	70

#### 4. Теми лабораторних занять

№ з/п	Назва теми	Кількість Годин
1	Алгоритми. Імперативний підхід	6
2	Мова програмування Python. Константи, змінні, типи даних. Оператори і вирази. Порівняння МП Fortran та Python.	6
3	Оператори управління МП Python.	6
4	Функції	6
5	Рекурсивні описи функцій.	6
6	Форми рекурсії.	4
7	Модулі	8
8	Структури даних. Список. Об'єкти і класи. Кортеж. Словник. Послідовності. Множини. Посилання. Рядки.	6
	<b>Усього:</b>	<b>48</b>

#### 5. Завдання для самостійної роботи

№ з/п	Назва теми	Кількість Годин
1	Алгоритми. Імперативний підхід	2
2	Про поняття алгоритма. Декларативний підхід	6
3	Мова програмування Python. Константи, змінні, типи даних. Оператори і вирази. МП Fortran.	7
4	Оператори управління МП Python.	8
5	Функції. Програмування процедур в Fortran.	4
6	Рекурсивні описи функцій.	8
7	Форми рекурсії.	8
8	Модулі	7
9	Структури даних. Список. Об'єкти і класи. Кортеж. Словник. Послідовності. Множини. Посилання. Рядки.	6
10	Класи. Методи. Наслідування. Введення-виведення. Файли Виключення. Обробка виключень. Стандартна бібліотека.	6
11	Графи. Алгоритми обробки графів. Бінарні дерева. Швидкі алгоритми сортування.	8
	<b>Усього:</b>	<b>70</b>

#### 6. Індивідуальні завдання

Індивідуальні завдання надаються по закінченню вивчення першого розділу, відповідають тематиці лабораторних робіт.

### 7. Методи контролю

1. Поточний контроль – виконання тестових завдань (з використанням комп'ютера). Завданням поточного контролю є перевірка розуміння та засвоєння певного матеріалу, вироблених навичок складання алгоритмів та програм, умінь самостійно розробляти алгоритми та програми, здатності осмислити зміст теми чи розділу.

*Об'єктами поточного контролю* знань студента є:

- Систематичність, активність роботи, виконання завдань для лабораторних робіт;
- виконання завдань для самостійного опрацювання;
- виконання індивідуального завдання.

2. Підсумковий контроль – виконання тестових і практичних завдань зі складання алгоритмів та програм; розв'язування проблемних питань; виконання творчих завдань, тестовий контроль різних ступенів складності. Завданням підсумкового контролю є перевірка розуміння студентом програмного матеріалу в цілому, логіки та взаємозв'язків між окремими розділами, здатності творчого використання накопичених знань, уміння сформулювати своє ставлення до певної проблеми навчальної дисципліни тощо.

### 8. Схема нарахування балів

Поточний контроль, самостійна робота, індивідуальні завдання											Індивідуальне завдання	Разом	Підсумковий семестровий контроль (чотирирівнева шкала)	Сума
Розділ 1		Розділ 2						Розділ 3						
T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11				
5	5	5	5	5	5	5	5	5	5	5	5	60	40	100

T1, T2 ... – теми розділів.

### Шкала оцінювання

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка за національною шкалою
	для чотирирівневої шкали оцінювання
90 – 100	відмінно
70-89	добре
50-69	задовільно
1-49	незадовільно

### 9. Рекомендована література

#### Основна література

1. Крєневич А. П. Python у прикладах і задачах. Частина 1. Структурне програмування Навчальний посібник із дисципліни "Інформатика та

- програмування" [Електронний ресурс] / А. П. Крєневич. – 2017. – Режим доступу до ресурсу: [www.matfiz.univ.kiev.ua/books](http://www.matfiz.univ.kiev.ua/books).
2. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов.– М.: Мир, 1979.– 536 с.
  3. Васильев А. Н. Python на примерах. Практический курс по программированию / А. Н. Васильев. – СПб.: Наука и техника, 2016. – 432 с.
  4. Н.Г. Кокодий, В.А.Свич, "Математическое моделирование в физике", – ХНУ имени В.Н. Каразина, 2007 г. – 328с.
  5. Н. Вирт. Алгоритмы + структуры данных = программы. – М.: Мир. - 1985.
  6. Соколов О. Ю., Зарецька І. Т., Жолткевич Г. М., Ярова О. В. Інформатика для інженерів. – Харків: Факт. – 2005.

#### **Допоміжна література**

1. Збірник задач з дисципліни "Інформатика і програмування" Вакал Є.С., Личман В.В., Обвінцев О.В., Бублик В.В., Довгий Б.П., Попов В.В. -2-ге видання, виправлене та доповнене –К.: ВПЦ "Київський університет", 2006.– 94с.

#### **10. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення**

1. С. Шапошникова Основы программирования на Python <http://younglinux.info>
2. A Byte of Python (Russian), Версия 2.01  
[https://github.com/swaroopch/byte\\_of\\_python](https://github.com/swaroopch/byte_of_python)